Advanced Diagnostics and Prognostics Testbed (ADAPT)

Dougal Maclise[*], Andre Goforth[*], Adam Sweet[**], Dwight Sanderfer[*], Joe Camisa[*], Gerald Temple[*], David Nishikawa[*], Eric Barszcz[**], Leonard Olson[**], Serdar Uckun[*]
NASA Ames Research Center, Moffett Field, CA 94035[*]
QSS Group Inc./NASA Ames Research Center, Moffett Field, CA 94035[**]

Introduction

The Advanced Diagnostics and Prognostics Testbed (ADAPT), currently in its final testing at the Ames Research Center prior to full operations, offers a unique opportunity for high fidelity testing of both flight hardware systems and ISHM diagnostic / prognostic software. ADAPT offers several significant advantages because of its unique concept of operations and its highly flexible systems design. For example, the ADAPT architecture provides for an "Antagonist" role, which ultimately represents an independent and realistic source of faults broadcast throughout the system and made available to the diagnostic / prognostic software as well as engineers playing other roles in the test scenario.

Background

ISHM technologies must buy their way onto space vehicles because their non-negligble footprint of power, weight, communications, etc. competes at the margin with other vehicle components' corresponding needs. Justifications for insertion of such technologies rest mainly on lowering overall system cost for a given set of mission requirements where risk, system reliability and cost have to be balanced. Thus, for a given set of requirements a significant challenge for any new vehicle program is deciding how much and what kind of ISHM technology to use that will maximize its return on investment (ROI).

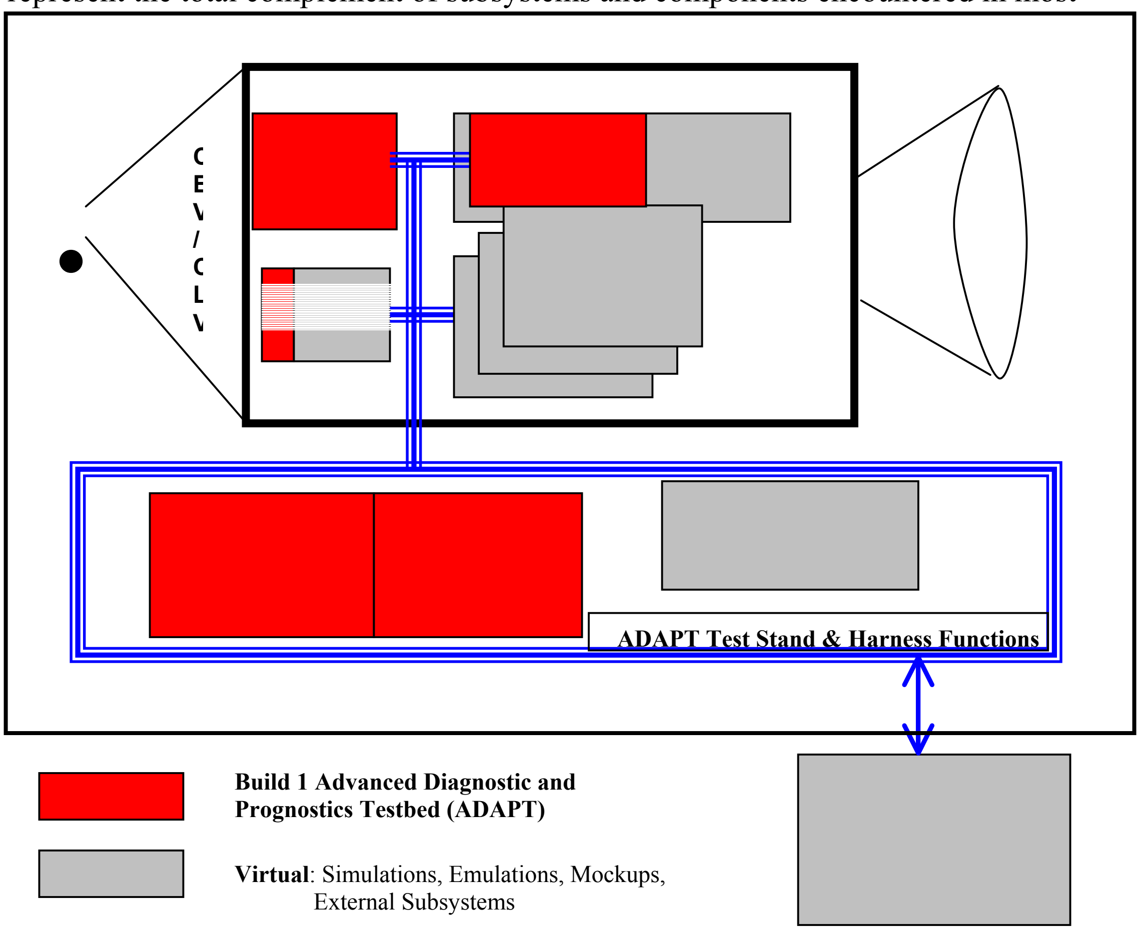
The ADAPT project is currently completing the first block of its development, Build 1, which in part will allow full verification and validation of three key operational roles, whose interactions are integral to the ADAPT Operations Concept. These roles will be described below.

Concept of Operations

Below is a block diagram of the major components of the testbed that illustrates their interaction. The outer block represents the testbed as a facility. This facility encapsulates a number of components some of which normally will be provided by users and the remainder will typically be indigenous to the facility in that they provide control, testing and recording faculties external to the customer's system. The elements that can be provided by the customer may be one or more of the blocks depicted in the diagram as internal to a flight vehicle illustrated here as CEV/CLV. In the block diagram, the CEV/CLV occupies the upper half of the large rectangle representing the ADAPT facility.

ADAPT Test Stand & Harness Functions The indigenous faculties of the testbed include the ADAPT Environment Generator, the Test Recording System, the Fault Injection System and the Health Management Engineer. The second and third systems feature console positions, the Observer and Antagonist consoles. These consoles along with the Environmental Load function provide interfaces for controlling the testbed and observing the test bed parameters as well as observing the behavior of customer's test articles. The latter may include diagnostic tools or health management hardware controllers or subsystems. Note that for the most part these functions are not part of a flight system, rather they serve as a bird's eye view of how the customer's equipment reacts in a given test scenario. There is another testbed function that is not shown within the ADAPT Test Stand & Harness Functions block because often it is an actual part of a flight system. This function is called the Health Management (HM) Engineer and is shown as being within the CEV/CLV. Although in this paper it is represented as being part of the customer's flight vehicle, in the current ADAPT prototype it is treated as a peer with the other operational roles and therefore has its own console position called the User. It is important to note that the HM Engineer function may be implemented in a variety of ways depending on the customer's needs. In terms of where it belongs in the ADAPT concept of operations we have found it useful to view it as having dual roles: in one it is placed within the customer's vehicle, in the other it is part of the testbed functions. The decision as to where the User best fits in the ADAPT architecture will be made on the basis of subsequent operational experience and on a case by case basis analysis of each customer's requirements.

CEV/CLV—Adaptable Testbed Architecture The testbed is highly adaptable and may be configured to mimic a wide variety of specific aerospace systems. As a starting point our approach is to use less expensive generic space-like and flight-like equipment that, as a whole, may represent the total complement of subsystems and components encountered in most



ADAPT Facility Concept of Operations

aerospace systems, such as the Shuttle, the International Space Station (ISS), robotic probes and the Crew Exploration Vehicle (CEV). In some cases the incorporation of flight article components in experimental runs may be achieved through remote access. This is represented in the block labeled External Partners & External Testing. The extent that hardware flight articles such as power or propulsion systems, for example, may be included within the facility is limited by budget and safety considerations.

In practice, customers are likely to want to limit evaluation and testing to specific subsystems or sub elements of these because of cost limitations or difficulties in providing flight hardware or its equivalent.

Live Test Environment

The ADAPT test environment is designed to support more that just standard test cases and scenarios. In fact even a fixed test set-up can be operated in an arbitrary number of ways since the faults introduced into the system during the test need not be determined in advance. The ADAPT Live Test Environment offers the opportunity for three distinct, independent roles to participate in any given test. These three roles, which can be staffed by individuals or by larger teams, are the "User", the "Antagonist", and the "Observer". Perhaps the most unique feature of ADAPT's test environment is the antagonist subsystem that includes unscripted software and hardware generated faults and anomalies. In addition, ADAPT supports physical faults generated through manually initiated destructive test cases. Such testing will offer a high level of test case signature fidelity with which to evaluate and compare candidate ISHM technologies. Throughout the test, the "User" plays the part of the Health Management Engineer, who in this role is ignorant of both the Antagonist and any role being played by diagnostic / prognostic software. He or she observes faults as they arise, and attempts to address them with his or her own resources. The nature of the interaction of the user with the system and the resources available to the "User" depend on the customer's mission operations scenarios where the Health Management Engineer role may be part of another crew interface or may be separate.

The "Observer" is independent of the other two roles. His or her responsibility is to accurately document the entire stream of data, including the activities of the individuals occupying the other two roles, as well as the results from any software that may be in the loop.

Test Metrics

One of the assets of ADAPT will be the development of libraries of test scenarios, test cases, technical performance metrics (TPM), figures of merit (FOMS), etc. for the customer's test articles. One of the goals of ADAPT is to serve as a "Consumer Reports" to NASA's and industry's ISHM community by collecting and reporting the TPMs and FOMs measured for each test article. The following are examples of TPMs and FOMs that ADAPT may collect. As ISHM systems can vary widely in their underlying approaches, not all of these will necessarily be collected for each test article, and additional FOMs not listed below may be implemented for particular test articles:

- **Health Management Development Cost** – measured in hours and material dollars. This measure will indicate the number of person-hours and dollar cost of materials that were spent to apply the ISHM technology to the ADAPT test subsystems for the evaluation test. This cost includes cost to procure the technology and any application or test equipment, the hours spent in developing subsystem models, writing code specific to the application and testing the specific application to prepare the technology for the evaluation test. It does not include the cost to refine, improve or otherwise develop the technology itself.

- **Prognosis to Criticality** – measured in time (msec., sec., min. or hr.). This measure will indicate the time difference between the Time to Criticality and the Time to Remediate a failure. The Time to Criticality is defined as the time from the indication of a fault or degradation of a function to the complete failure of that function. An example is the time from the detection of a hydrogen leak in a tank to the explosive combustion of the hydrogen. The Time to Remediate is the time that the system needs to prevent the catastrophic failure from the first detection of the fault. This measure is obviously application specific. This measure may also include a measure of the confidence of the fault detection and the prediction of Time to Remediate.
- **Fault Detection Coverage** – measured in percentage. This measure will indicate the ratio of the number of faults that the ISHM subsystem can detect to the total number of faults that the system or subsystem can experience (those identified in the Fault Tree analysis or FMECA). The Fault Detection Coverage requirements will be broken down into Failure Criticality categories.
- **Fault Isolation Rate** – measured in percentage for an Ambiguity Group Size. This measure will indicate the ratio of the number of faults that can be detected and isolated to a single, or number of, Line-Replaceable-Units (LRUs), that is equal to or smaller than the specified group size, to the total number of faults that can be detected. For example, a Fault Isolation Rate of 90% for an Ambiguity Group of 3 means that 90% of the faults that can be detected can be isolated to a group of 3 or fewer candidate LRUs responsible for the fault.
- **False Alarm Rate** – measured in percent. This measure will indicate the ratio of the number of times that an ISHM technology will indicate a fault or failure of the test subsystem when there is no fault or failure with function of the subsystem to the number of faults that can be detected. This will include false indications of faults and true indications of faults that are not correctly isolated. This will not include true faults that are not indicated. (These are part of the Fault Detection Coverage measure.)
- **Fault Remediation Time** – measure in time (msec., sec., min. or hr.). This measure will indicate the time from detection of a fault to the completion of remediation of the fault. The remediation of the fault may include shutdown of that subsystem, replacement of the failed LRU, or removal of the need for that function. This will often include the time for manual intervention in the subsystem. This particular FOM will be an indication of the merit of not just the ISHM technology, but also the subsystem design and remediation strategies employed.

Another testing goal is to determine the TPMs below as well as others:

- **Fault Detection Time –** measured in time (msec., sec., or min.). This measure will indicate the time from the occurrence of fault, anomaly or functional degradation to the detection of the event. This measure will be specific to the given fault. For a specific ISHM technology, these measures may be averaged over the whole range of faults, grouped by percentiles, or grouped by fault criticality. This measure will be highly dependent on the processing capability of the hardware running the ISHM technology.
- **Fault Isolation Time** – measure in time (msec., sec., or min.). This measure will indicate the time from the detection of a fault, anomaly or functional degradation to the indication of the responsible LRU(s). This measure will be highly dependent on the processing capability of the hardware running the ISHM technology.
- **Time to Isolate Faulty LRU** – measured in time (msec., sec., min. or hr.) This measure will indicate the time from indication of a detected fault to the isolation of the fault to the correct LRU or LRUs. This differs from Fault Isolation Time in that the final isolation is to the correct part or parts responsible for the fault indication, not just the group of candidates.
- **Missed Alarms** – measured in percent, ppt (parts per thousand), or ppm (parts per million). This measure will indicate the number of faults not detected by the ISHM technology divided by the total number of faults injected. This measure will be highly dependent on the selection of subsystem sensors, the reliability of those sensors, the design of the avionics hardware, as well as the application of the ISHM technology.
- **False Alarms –** measured in percent, ppt or ppm. This measure will indicate the number of false indications of a fault divided by the total number of faults injected.

- **False Isolations –** measured in percent, ppt or ppm. This measure will indicate the number of detected faults that cannot be isolated to five or fewer candidates responsible for the fault or are incorrectly isolated to groups of candidates that are not responsible for the fault.
- **Memory Footprint –** the maximum amount of memory used by the application through all tests.
- **CPU load –** the amount of CPU processing power used by the application, sampled at intervals during the test run.

In Build 1 the focus will be on establishing a set of FOMS for the first test article, the diagnostic reasoner *HyDE*, to be evaluated by the testbed. The definitions of these FOMS are given below. For this paper the definition of a test run is as follows: A test run is a period of time which starts with the testbed in a known initial state and is operated for a given time period in which nominal operations are performed by the "User" and which may include phenomena injected by the "Antagonist".

• **Correctness of Fault Detection –** This measure will be classified into three categories:

correct detection: If either of the following two (remaining) cases is true, the test run shall be classified as correct detection: (1) If a fault is not injected in the test run and the ISHM system does not report that a fault occurred during the test run, or (2) If a fault is injected in the test run and the ISHM system reports that a fault occurred after the fault was injected.

false alarm: If either of the following two cases is true, the test run shall be classified as a false alarm: (1) If a fault is not injected in the test run, but the ISHM system reports that a fault occurred during the test run, or (2) If a fault is injected in the test run and the ISHM system reports that a fault occurred before it was injected.

missed alarm: If the ISHM system does not report a fault during a test run where a fault was injected, the test run will be classified as a missed alarm.

• **Correctness of Fault Isolation –** This measure will be classified into two categories:

correct isolation: If a test run is classified as a correct detection, and the ISHM system reported the injected fault as part of any ambiguity group at any point after the fault injection, the test run shall be classified as a correct isolation.

incorrect isolation: If a test run is classified as a false alarm or missed alarm above, it shall be also classified as incorrect isolation. Also, if a test run is classified as a correct detection, but the ISHM system did not report the injected fault as part of any ambiguity group at any point after the fault injection, the test run shall be classified as an incorrect isolation.

• **Fault Detection Time –** This measure will only be calculated for test runs that are determined to be correct detections. The fault detection time shall be calculated as the difference between the time at which the fault was injected to the time at which the ISHM system reported a failure, given in the most convenient common time units (sec, ms, min). As this measure will be highly dependent on the processing capability of the hardware running the ISHM technology, the CPU speed and the amount of memory available to the host computer will also be given.

• **Fault Isolation Time** – This measure will only be calculated for test runs that are determined to be correct isolations. The fault isolation time shall be calculated as the difference between the time at which the fault was injected to the time at which the ISHM system reported the fault isolation or ambiguity group, given in the most convenient common time units (sec, ms, min). As this measure will be highly dependent on the processing capability of the hardware running the ISHM technology, the CPU speed and the amount of memory available to the host computer will also be given.

• **False Alarm Rate** – The number of test runs classified as false alarms divided by the total number of test runs.

• **Missed Alarm Rate** – The number of test runs classified as missed alarms divided by the total number of test runs.

Future Plans

Once the “Consumer Reports” to the ISHM community role is established by the end of Build 1, we will add the role of serving as an ISHM “Underwriters Laboratory” facility. To do this we will finalize a baseline of FOMs and TPMs that are supportable within ADAPT’s capabilities and work with the ISHM community to establish accepted regimes for testing. The rest of our plans for expansion of ADAPT’s capabilities fall in the following areas: hardware, software, test scenario development and advanced concepts for mission operations.

Hardware Build 1 hardware includes switch-able solar and battery power sources. In a subsequent build we will add fuel cell sources to ADAPT’s representative implementation of a generic spacecraft power subsystem.

Software Build 1 software supports evaluation and testing of a particular ISHM technology type. Our plans are to implement a software architecture that can accommodate a broader collection of technologies and that will support combinations of ISHM technologies operating and coordinating together across different phases such as prelaunch checkout, launch, flight, re-entry and post flight checkout.

Advanced Concepts for Mission Operations To provide the customer a rich set of design alternatives for developing the Health Management Engineer or “User” console, we plan to interface with the Intelligent Spacecraft Interface Systems (ISIS) and the Mission Control Technologies laboratories at Ames and work with these labs to identify and refine the customer’s requirements in how health management is integrated into mission operations.